Oracle® AutoVue

AutoVue Web Client Guide

Release 21.1.0.4

F86931-04

April 2025

ORACLE®

AutoVue Web Client Guide, Release 21.1.0.4

F86931-04

Contents

Preface

The Oracle AutoVue 3D Professional Installation Guide describes how to install and configure Oracle AutoVue and its associated components.

For the documents from previous releases, go to the AutoVue Documentation Web site on the Oracle Technology Network (OTN) at
https://www.oracle.com/technical-resources/documentation/autovue.html

Related Documentation

For more information, see the following documents in the Oracle AutoVue documentation set:

- *Oracle AutoVue Installation and Configuration Guide*
- *Oracle AutoVue Viewing Configuration Guide*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Introduction

Oracle AutoVue is used for viewing 3D Computer-Aided Design (CAD) drawings created across a variety of CAD authoring software.

1.1 AutoVue Client

The AutoVue client is available as a web-based application in addition to the client/server deployment.

1.1.1 AutoVue Deployment Components (Client Components)

AutoVue client components need to be hosted within an application server or a Web server. The client components need to be configured to communicate with the VueServlet, which in turn communicates with the AutoVue server. You can connect to AutoVue using the URL served by this application or Web server. You can deploy the AutoVue client components in a an environment integrated with a document management system or in a non-integrated environment.

For non-integrated environment, AutoVue is deployed as a single page web application. The main page, that is the index.html page has reference to the AutoVue component which is responsible for loading every feature in the page, such as creating session, creating, opening and closing file, fetching the SCS data and many more.

When AutoVue is integrated with a document management system, you must copy over the required WAR(filesys.war) and online help files to the application server that hosts the integration components and/or the document management system.

2

AutoVue System Requirements

This chapter provides the system requirements for AutoVue release.

2.1 System Requirements

The web client of AutoVue works on different platforms such as Windows, Linux, Mac.

Here are the system requirements for the AutoVue release.

2.1.1 Server Platforms

Let's look at the server platforms that are certified for AutoVue installation and/or hosting.

Table 2–1   Oracle-Certified Operating Systems

Operating System	Versions
Windows (AutoVue running in 64-bit mode)	▪ Windows Server 2022 ▪ Windows Server 2019
Oracle Linux (AutoVue running in 64-bit mode)	▪ 9.X (x86_64) ▪ 8.X (x86_64) ▪ 7.X (x86_64)
Red Hat Enterprise Linux (AutoVue running in 64-bit mode)	▪ 9.X (x86_64) ▪ 8.X (x86_64) ▪ 7.X (x86_64)

2.1.2 Client Platforms

Let's look at the platforms that are certified for the AutoVue client.

Table 2–2   Client Platforms

Platforms	Browser
Windows 11 — 64-bit Windows 10 — 64-bit	▪ Firefox - ESR 128.9[1] ▪ MS Edge - 135[1] ▪ Chrome - 135[1]
Apple Mac OS 14.3.1 Apple Mac OS 13.6.4	▪ Safari - 17.3.1[1] ▪ Safari - 16.6.1[1] ▪ Firefox - ESR 128.9[1]

Table 2–2   (Cont.) Client Platforms

Platforms	Browser
Java Virtual Machine	⊠ Java SE 8 update 461 64-bit

[1]   See the Oracle Software Web Browser Support Policy:
https://www.oracle.com/middleware/technologies/browser-policy.html

2.1.3 Application Servers

Here are the Oracle-certified application servers:

Table 2–3   Application Servers

Platform	Version
WebLogic	12cR2

Tomcat	9.0.104, 10.1.40, and 11.0.6
	
	Note: If you are using Tomcat 10 or Tomcat 11 to deploy VueServlet, add the following in conf\Catalina\localhost\VirtualStore.xml or conf\context.html:
	
	<Context>
	
	  ...
	
	  <Loader jakartaConverter="TOMCAT" />
	
	</Context>

Jetty	9.4.57

2.1.3.1 Response Headers

Add the following headers to the application server:

Table 2–4   Response Headers

Headers	Value	Description
X-XSS-Protection	1; mode=block	Cross-site scripting instruction
x-frame-options	DENY	Used to avoid click-jacking attacks.
Access-Control-Allow-Methods	GET, POST	Specifies (http) methods allowed while accessing a resource.
Strict-Transport-Security	max-age=31536000; includeSubDomains; preload	Defines whether to use encrypted/non encrypted access.
x-content-type-options	nosniff	Helps avoid mime type sniffing.
Access-Control-Allow-Headers	Origin, X-Requested-With, Content-Type, Accept	Indicates which http headers can be used.
Cache-Control	private	Controls caching in browser for requests and responses.
Access-Control-Allow-Origin	<protocol>://<server-domain>:<port>	Indicates whether the response can be shared with requesting code from the given origin.

2.1.4 Integrations

This AutoVue release is integrated with demo sample FileSys.

3

AutoVue Components

You can use AutoVue either as a standalone application or as an application integrated with a document management system.

Let's look at the several components of AutoVue:

- The AutoVue server

- An application server hosting the VueServlet and document management system.

- A Web server or an application server hosting AutoVue client components

- AutoVue client

3.1 AutoVue Components

This section discusses the AutoVue client components. AutoVue can be closely integrated with different document management systems.

Figure 3–1   Integrated Deployment
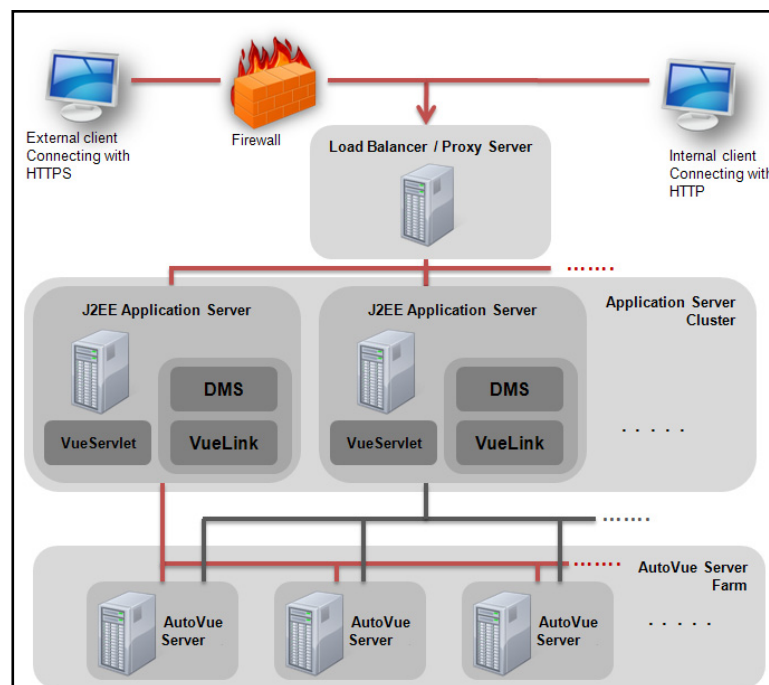
In this scenario:

- Internal and external clients connect to a load balancer

- A load balancer routes requests to the VueServlet which is deployed in an application server cluster

- The VueServlet directs requests to an AutoVue server that is deployed in a server farm.

- The AutoVue server in turn communicates with the VueLink and the VueLink communicates with the document management system to fulfill all the request.

- The AutoVue server entries in the VueServlet configuration must be identical across all the VueServlet instances.

4

Install AutoVue

This chapter provides instructions for installing AutoVue.

4.1 Linux Prerequisites

To correctly install AutoVue on a Linux OS, it is recommended that you have basic knowledge of Linux and its administration.

1. To install packages on a Linux system you must have appropriate administrative privileges.

2. Wine must be installed.

   > **Note:** See Instructions for Installing Wine for additional information about versions and ways of installing Wine.

3. For AutoVue installation on Linux, if the installer cannot find the JRE installed, the following command is recommended:

   ```
   ./InstallClientServer_lin.bin LAX_VM "<path to java>"
   ```

4. This step is optional. Install the corefonts package in environments where Microsoft Office Document support is an important concern. Install the TrueType core fonts package from https://corefonts.sourceforge.net/. Note that the AutoVue server must be restarted for the changes to take effect.

   Verify that the fonts package is installed and configured properly by checking that the directory /usr/share/fonts/msttcorefonts exists and contains a collection of TrueType (TTF) files.

   > **Note:** The following step needs to be performed after the AutoVue servers have been installed:
   >
   > ▫ Create symbolic links from the TTF files in the /usr/share/fonts/msttcorefonts to the <AutoVue Install Dir>/jvuew_c/windows/fonts directory.
   >
   > ▫ `-cd <AutoVue Install Dir>/jvuew_c/windows/fonts`
   >
   >   `-ln -s /usr/share/fonts/msttcorefonts/*.ttf`

> **Note:** Fonts are governed by certain licensing restrictions. Ensure that you verify the licensing for fonts before copying them over to different machines.

4.1.1 Installation Instructions for Wine

The AutoVue server has a dependency on the wine package. In releases prior to version 21.1.0, AutoVue had a specialized version of wine that was distributed from the oss.oracle.com website.

In version 21.1.0.x, the AutoVue server still has the wine dependency. Use the following table to decide the version of WINE you would require:

Table 4–1 Linux version and compatible WINE version

Linux Version	WINE Version
7	4.x
8	6.x
9	8.x

> **WARNING:** Don't use the wine-av-package that was used in previous releases of AutoVue. The old packages contained 32-bit executables, while version 21.1.0 is 64-bit.

Installation Instructions for Oracle Linux

Following are the instructions for installing wine and its pre-requisite packages on Oracle Linux:

1. Enable CodeReady Builder. This step applies to Oracle Linix 8 and Oracle Linux 9 only:

   ```
   $ sudo subscription-manager repos --enable
   codeready-builder-for-rhel-<x>-$(arch)-rpms
   ```

2. Install the Extra Packages for Enterprise Linux (EPEL) repository:

   ```
   $ yum-config-manager --enable ol<x>_developer_epel
   ```

3. Install Wine:

   ```
   $ sudo yum install wine
   ```

   The yum command should confirm the installation of Wine when it is successful.

   > **Note:** Replace <x> with the Linux major version 8 or 9.
   >
   > For Oracle Linux 7, you need to resolve any dependencies manually.

Installation instructions for Red Hat Enterprise Linux (RHEL)

Following are the instructions for installing wine and its pre-requisite packages on Red Hat Enterprise Linux:

1. Enable CodeReady Builder. This step applies to RHEL8 and RHEL9 only:

   ```
   $ sudo subscription-manager repos --enable
   codeready-builder-for-rhel-<x>-$(arch)-rpms
   ```

2. Install the EPEL RPM:

```
$ sudo yum -y install
https://dl.fedoraproject.org/pub/epel/epel-release-latest-<x>.noarch.rpm
```

3. Install wine:

```
$ sudo yum install wine
```

The yum command should confirm the installation of Wine when it is successful.

> **Note:** Replace <x> with the Linux major version 8 or 9.
>
> For RHEL7, you need to resolve any dependencies manually.

> **Note:** For more details, consult KM Note: Doc ID 2990780.1

4.2 Install AutoVue Client Components

Let's look at the two ways of installing AutoVue client components.

1. Integrated Deployment: Deploy the AutoVue client component in an environment that is integrated with a Document Management System. The AutoVue web client is integrated with FileSys.

   AutoVue can be integrated with various document management systems using the Integration Software Development Kit (ISDK).

2. Non-integrated Deployment: Deploy the AutoVue client component as a stand-alone application that is a non-integrated deployment.

4.2.1 Install AutoVue in an Integrated Environment

This section describes how to install AutoVue in an environment that is integrated with the Filesys document management system.

> **Note:** For the FileSys integration to work, it's important that the FileSys document management system and the AutoVue VueServlet have to be deployed on the WebLogic application server.
>
> This project is in the `\AutoVueIntegrationSDK\FileSys` folder.

Here's how you install AutoVue in an integrated environment.

1. Install AutoVue through the installer. For more details refer to the AutoVue Client/Server Deployment Installation Guide.

2. Next, to install FileSys, extract the contents of the AutoVueIntegrationSDK.zip. The Filesys folder is located in the AutoVueIntegrationSDK /Repository folder. The FileSys folder contains the following sub folders:

   - The /WebApplication folder that contains a filesys.war file and a /filesys folder. The content in the /filesys folder is the unzipped version of the filesys.war.

   - The /Repository folder contains filesysRepository.zip which contains sample files used by the Sample Integration for Filesys.

- The /ESAPI_Resources folder contains the OWASP Enterprise Security API properties files: AvESAPI.properties and validation.properties.

- The /OEVF folder contains two GUI files used for the OEVF demo.

3. Extract the "filesysRepository.zip" that is present in \AutoVueIntegrationSDK\FileSys\Repository and move its contents directly in the Repository folder.

4. In case, you need to copy the contents filesysRepository.zip to any folder of your choice or you need to configure 3D samples from any other folder.

   a. Create a folder filesysRepository and add the 3D samples in the folder. For example, the filesysRepository folder path is: C:\filesysRepository.

   b. Use this path to configure *RootDir* in the WEB-INF\web.xml file.

5. Make the code changes to WEB-INF\web.xml of the FileSys directory or the filesys.war in \AutoVueIntegrationSDK\.

```
<!--  context parameters are available to all servlets -->

    <context-param>
      <param-name>RootDir</param-name>
      <param-value>C:\<company_name>\AutoVueIntegrationSDK\FileSys\Repository\
      </param-value>
    </context-param>
```

Note: Note: C:\<company_name>\AutoVueIntegrationSDK\FileSys\Repository\ - Repository folder path with the sample 3D/2D files.

```
================================================================================
=============================================
<!-- This URL is only needed to construct thumbnail URLs -->
  <context-param>
    <param-name>RootURL</param-name>
    <param-value>C:\<company_name>\AutoVueIntegrationSDK\FileSys\Repository\
</param-value>
  </context-param>
================================================================================
==============================================
<!-- This is Autovue VueServlet endpoint (Default value will be empty and
pointing to the same domain,
Make changes here If server domain and port is changing)  -->
<context-param>
    <param-name>JVueServer</param-name>
    <param-value><server-domain>:<port></param-value>
  </context-param>

<!-- This is support formats list for Web client -->
    <param-name>Formats</param-name>
  <!--  Comma separated list of formats -->
  <param-value>.jt,.CATPart</param-value>
</context-param>

================================================================================
=============================================
```

> **Note:** If HTTPS setup has to be made, update the EnableSSL to true.

```
<init-param>
     <param-name>EnableSSL</param-name>
     <param-value>0</param-value>
</init-param>
```

> **Note:** For more information on SSL, refer to the "Enabling SSL Communication" section of the Oracle AutoVue Client/Server Deployment Security Guide.

6. Copy the \AutoVueIntegrationSDK\FileSys\ESAPI_ resources in the same folder as your project or any path of your choice.

7. Set ESAPI_resources location in Java Options.

   In the WebLogic start options in the location C:\<company_name>\Middleware\<company_name>_Home\user_projects\domains\base_domain\bin\setDomainEnv.cmd/sh

   Set the Java Options as shown in the example:

```
JAVA OPTIONS
set "JAVA_OPTIONS=%JAVA_OPTIONS%
-Dcom.cimmetry.vuelink.esapi.resources=C:\Oracle\AutoVueIntegrationSDK\FileSys\
ESAPI_resources"
```

8. Update the credential.txt file in location filesys/WEB-INF/lib for user names and passwords in the format username: password:

9. In the ESAPI_resources folder, there is a AvESAPI.properties file. Ensure that the 3D file format extensions you want to view in AutoVue are included in the allowed list as defined by HttpUtilities.ApprovedUploadExtensions.

10. If you are using a WebLogic application standalone server, Deploy FileSys directory or filesys.war based on the place where you have changed the web.xml mentioned in step 5.

11. After you have deployed the filesys.war in your application server, test the VueServlet by accessing the URL to the VueServlet – `http://host:port/filesys/servlet/VueServlet`.

12. Run jVueServerX.exe (/bin) to start the AutoVue server.

13. Use the URL to login to FileSys demo: `http://host:port/FileSys/jvue/jVue.html`.

> **Note:** It's important to clear the cache or cookies of the browser in case you've tested filesys before.

4.2.2 Install AutoVue in a Non-Integrated Environment

Here's how you install AutoVue in a non-integrated environment.

1. Install AutoVue through the installer. For more details, refer to the Oracle Autovue Client/Server Deployment Installation and Configuration Guide.

2. After installation, navigate to the <Autovue-Install Root>\html5, and select html5.

   a. Do changes in index.html and main.js files inside html5 folder by following the steps in the section Running AutoVue Web Application as Stand-Alone.

        b.   Create a war file of the html5 folder.

        c.   Deploy the .war file in application server (for example weblogic).

3. Deploy VueServlet WAR on the application server. To deploy VueServlet on Application Servers, see "Deploying the VueServlet on Application Servers" in Oracle AutoVue Installation and Configuration Guide.

4. Run **jVueServerX.exe** (/bin) to start the AutoVue server.

5

Integrating AutoVue in a Document Management System Environment (Filesys)

AutoVue can integrate with various document management systems. You must factor the integration component into your deployment planning. If you are building your own integration or using a third-party integration with AutoVue. Refer to the Oracle AutoVue Integration Guide.

After you complete the integration of AutoVue with the Document Management System, the web application bundle has to be copied to your application webserver.

Here's how you can use the AutoVue component in the webpage.

1. Steps for Uptake

   a. Copy js and css folders from the <ISDK Installation Folder>\FileSys\WebApplication\filesys\jvue folder, and paste it in the location of your choice.

   b. frmAV.jsp file is available at location: <ISDK Installation Folder>\FileSys\WebApplication\filesys\jvue

   i) Use this file as a reference file and copy the file contents to the uptaking application view file.

   ii) Change the Javacript files location to point to the js folder location in step 1.

   iii) Modify the requirejs.config so that it points to the js folder location

2. Passing data to AutoVue Web Application

   In an integrated environment all these values are passed through two API's through the autovuehtml.js to the AutoVue component (as shown in the frmAV.jsp for example),

   1. Initialisation of AutoVue: In the integrating file the following data has to be defined and passed for initialization:

      a. First Parameter - The VueServlet URL

      ```
      var autovueVueservlet="<%=vueservlet%>";
      ```

      b. Second Parameter - INIT_PARAMS variable is an object which defines username, dms vuelink url and document management system arguments (comma separated keys) as shown below needs to be defined and passed.

      ```
      var INIT_PARAMS = {};
      INIT_PARAMS["USERNAME"] = "<%=username%>";
      INIT_PARAMS["DMS"] = "<%=dms%>";
      INIT_PARAMS["DMSARGS"] = "DMS_ARGS1;DMS_ARGS2";
      INIT_PARAMS["DMS_ARGS1"] = "value1";
      ```

```
        INIT_PARAMS["DMS_ARGS2"] = "value2";
```

2. Set file url on file change: Pass arguments filename and fileUrl to the function setfile on every file change.

```
setfile(filename, fileUrl)
```

3. Running AutoVue web application as stand-alone application

The AutoVue client is the main entry point to AutoVue's capabilities. The AutoVue client components has to be made accessible to end-users at an application server or Web server location.

The instructions for deploying the client components vary depending on whether you have AutoVue integrated with a document management system or if you are using a non-integrated environment.

Let's look at the instructions on how to deploy AutoVue in a non-integrated environment:

1. After installing AutoVue, navigate to html5 folderin the path : <Autovue-Install Root>\html5. This folder contains - js, css, index.html and samples folder.

The html5/samples folder is used for keeping the 3D sample files.

2. In index.html, replace '%serverUrl%' with VueServlet URL.

3. As the samples folder is hosted along with the rest of the content, copy the hosted URL, and replace '%serverUrl%' by its value in the "url" attribute of the data object in the main.js file.

4. Create a war file of the html5 folder and deploy the war file.

5. To verify successful installation and integration, enter the following URL: http://<domain-name>:<port>/html5.

6

API Parameters

This chapter discusses the Application Programming Interface (API) that the AutoVue client is built upon in both the integrated and non-integrated environment.

6.1 Non-integrated Environment - JavaScript API for Web Client

The AutoVue UI component requires attributes in a non-integrated environment. Here are some of the attributes:

Table 6–1   APIs in Web Client

Name	Type	Description
filename	string	The document file URL
server-url	string	The VueServlet URL
user-name	string	username

6.2 Environment Integrated with Document Management System - JavaScript API for Web Client

The AutoVue UI component requires attributes in an environment integrated with document management system. Here are some of the attributes:

Table 6–2   APIs in Web Client

Name	Type	Description
filename	string	The document file name
server-url	string	The VueServlet URL
user-name	string	username
seed-url	string	Document Management System File Url
dms	object	Document Management System Properties Object

7

Formats

AutoVue solutions supports technical document types such as 3-D Computer Aided Design (CAD) and Electronic Design Automation (EDA).

7.1 Supported Functionality

With this version of AutoVue, you can view, manipulate and measure 3D files. Here are some of the formats and the versions that are supported for the AutoVue release:

Table 7–1   Formats Supported

Vendor Product/File Format	Up to Versions
AutoDesk Inventor	2025
CATIA V5	5-6R2024 (R34)
Creo Parametric	11.0
DirectModel (JT)	10.9
IGES	5.3, 5.2, 5.1
Parasolids	Upto version 37
SolidWorks	2025
Solid Edge	2025
STEP	AP 242, AP 214, AP 203 formats
STL	STL
Unigraphics	NX 2412 and NX 2406 Series 3D files

8

Configuration Options

AutoVue as a viewer displays designs or documents very close to their authoring applications. There may be slight differences in the AutoVue display as compared to the native applications. In some cases, there are configurations that can be performed to have AutoVue closely display the files as in the native applications.

Here's what these configuration options can control:

☒ How the file displays or prints in AutoVue

☒ The performance for loading the document

☒ The attributes to be displayed with the file

8.1 Viewing Configuration for 3D Files

After installing AutoVue, update the user ini file under the <AutoVue_Installation_ Dir>\bin\Profiles with the following default flags:

Table 8–1   INI Profile List

Options	Default Value
XFONTPATHS	<Provide AutoVue_Installation_Dir_path>\bin\fonts
NOACCELERATION	1
PMIINITIALVISIBILITY	1
DIGITSNUMBER	3
MEASUREMENTUNITS	1
DEFAULTFILEUNITS	5
INITIALVISIBILITY	1
PMITEXTRENDERINGSTYLE	1
PMITHRESHOLD	194
BKTYPE	BOTTOM
SHOWGLOBALAXES	1
EXPANDTREELEVEL	0

Table 8–2   [SMCOLORS] Options

Options	Default Value
BACKGROUNDCOLOR	8160676
SELECTIONCOLOR	15201010
SECTIONFILLCOLOR	6579455
SECTIONEDGESCOLOR	16711680

Table 8–3   [PMI] Options

Option	Default Value
COORDINATE_SYSTEM_ TREE_VIS	0
DATUM_FEATURE_ SYMBOL_TREE_VIS	0
DATUM_TARGET_TREE_ VIS	0
DIMENSION_TREE_VIS	0
FEATURE_CONTROL_ FRAME_TREE_VIS	0
LINE_WELD_TREE_VIS	0
LOCATOR_TREE_VIS	0
MEASUREMENT_POINT_ TREE_VIS	0
NOTE_TREE_VIS	
REFERENCE_GEOMETRY_ TREE_VIS	0
SPOT_WELD_TREE_VIS	0
SURFACE_FINISH_TREE_ VIS	0
WIRE_TREE_VIS	1

Table 8–4   [SMMeasures] Options

Option	Default Value
DENSITYUSE	0
DEFAULTDENSITY	1

Table 8–5   [SMUNITS] Options

Option	Default Value
DENSITYMASSUNIT	0
DENSITYLENGTHUNIT	2

A

Feedback

If you have any questions or require support for AutoVue please contact your system administrator.

If at any time you have questions or concerns regarding AutoVue, please contact us.

A.1 General AutoVue Information

Web Site	https://www.oracle.com/applications/autovue/
Blog	https://blogs.oracle.com/

A.2 Oracle Customer Support

Web Site	https://www.oracle.com/support/

A.3 My Oracle Support AutoVue Community

Web Site	https://community.oracle.com/hub/

A.4 Sales Inquiries

E-mail	https://www.oracle.com/corporate/contact/global.html